REMARKS

The Examiner has rejected Claims 1-3, 5-15, and 19-25 under 35 U.S.C. 103(a) as being unpatentable over Abraham et al., U.S. Patent No. 5,983,270, in view of Gleeson, U.S. Patent No. 5,627,829. Applicant respectfully disagrees with this rejection.

Specifically, with respect to Claim 1 et al., the Examiner continues to rely on col. 2, lines 31-60, and col. 7 of Abraham to meet applicant's claimed "data bus connecting the addressable memory unit and the first and second data processing units." Abraham, however, merely suggests one data processing unit, and thus inherently fails to suggest applicant's claimed "data bus connecting the addressable memory unit and the first and second data processing units" (emphasis added).

It is also noted that the Examiner relies on the disclosure of a "network" to meet applicant's claimed "data bus." Following are exemplary definitions setting forth the broadest plain and ordinary meaning of such terms, showing that a network simply does not meet applicant's claimed "data bus."

> "Network - A group of two or more computer systems linked together. There are many types of computer networks, including:
> 1 local-area networks (LANs) : The computers are geographically close together (that is, in the same building).
> 2 wide-area networks (WANs) : The computers are farther apart and are connected by telephone lines or radio waves.
> 3 campus-area networks (CANs): The computers are within a limited geographic area, such as a campus or military base.
> 4 metropolitan-area networks MANs): A data network designed for a town or city.
> 5 home-area networks (HANs): A network contained within a user's home that connects a person's digital devices. ... "
> http://www.webopedia.com/TERM/n/network.html
>
> "Bus - A collection of wires through which data is transmitted from one part of a computer to another. You can think of a bus as a highway on which data travels within a computer. When used in reference to personal computers, the term *bus* usually refers to *internal bus*. This is a bus that connects all the internal computer components to the CPU and main memory. There's also an expansion bus that enables expansion boards to access the CPU and memory. ... "
> http://www.webopedia.com/TERM/b/bus.html

Still yet, the Examiner now relies on col. 20, lines 51 - 67; and col. 21, lines 1 - 50 from Gleeson to make a prior art showing of applicant's claimed "second data processing unit

adapted to process incoming packets according to one of said plurality of instruction sets after the filtering, based on a thread assigned to the incoming packets by the first data processing unit." Moreover, the Examiner asserts that Gleeson teaches the use of a compression routine run after filtering that is called by the transmit manager thread which is assigned before filtering.

Applicant respectfully disagrees with such assertion. Whether the Examiner's statement above is true or not, applicant's claimed unit that "process[es] incoming packets according to one of said plurality of instruction sets" (emphasis added) is simply not disclosed by Gleeson. Specifically, the compression routine initiated by the manager thread in no way suggests use of one of said plurality of instruction sets, as claimed by applicant.

More importantly, Gleeson fails to even suggest the foregoing specifically claimed processing that is "based on a thread assigned to the incoming packets by the first data processing unit." First, Gleeson does not even suggest a first data processing unit (in addition to the second). Moreover, Abraham's filter engine (the first data processing unit, per the Examiner) in no way suggests any sort of selection of a thread to be executed by a second data processing unit, as claimed.

For these reasons, applicant's claim elements of Claim 1 are not met by the Examiner's proposed combination. Further, it would not have been obvious to one of ordinary skill in the art at the time of the invention was made to modify Abraham in view of Gleeson to use a second data processing unit adapted to process incoming packets according to one of said plurality of instruction sets after the filtering, based on a thread assigned to the incoming packets by the first data processing unit.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

Applicant respectfully asserts that at least the first and third elements of the *prima facie* case of obviousness have not been met, for the reasons set forth hereinabove. A notice of allowance or specific prior art showing of such features, in combination with the remaining claim elements, is respectfully requested.

It appears that the Examiner's rejection is still deficient with respect to many of applicant's dependent claims. For example, the Examiner relies on col. 7; col. 5, lines 46-67; and col. 6, lines 1-4 from Abraham to meet applicant's claimed:

"wherein at least one of said policies comprises:

a first address pointer element for identifying the location in said addressable memory unit of one of said plurality of instruction sets, and

a second address pointer element for identifying the location in said addressable memory unit of a state block" (see Claim 4 et al.).

Specifically, the Examiner argues that Abraham teaches an addressable memory unit of one of a plurality of instruction sets by virtue of database 72 (see Fig. 4 from Abraham below).

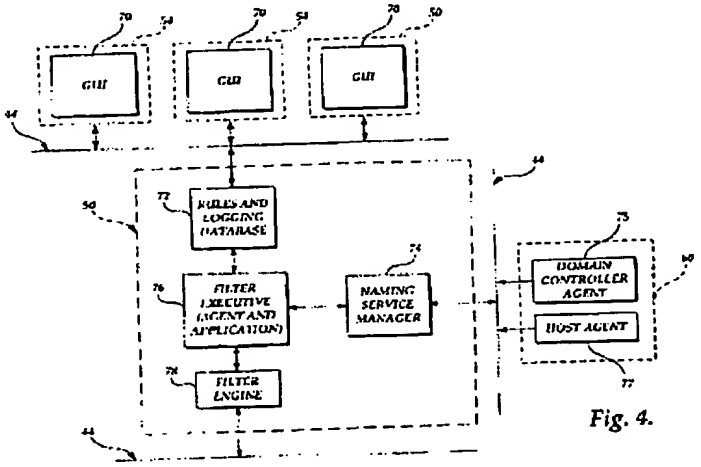

Fig. 4.

The Examiner continues by arguing that col. 5, lines 46-67; and col. 6, lines 1-4 from Abraham discloses applicant's claimed "location in said addressable memory unit of a state

block." Such excerpts, however, merely recite the functionality of the domain controller server 60 and client 54 (see above), which are not the database 72.

Thus, Abraham fails to even suggest an addressable memory unit with both instruction sets and a state block, as inherently claimed by applicant. Again, applicant teaches and claims "the location in said addressable memory unit of one of said plurality of instruction sets, and ... the location in said addressable memory unit of a state block." In order to meet applicant's claims, the database 72 would have to include the claimed state block, which it does not (as implicitly admitted by the Examiner).

Further, with respect to Claim 8, the Examiner also relies on col. 5, lines 46-67; col. 6, lines 1-4; and col. 7 from Abraham to make a prior art showing of applicant's claimed "wherein said data processing policy comprises a first address pointer to a starting address of a first set of instructions and a second address pointer to a starting address of a state block stored in said addressable memory unit, said state block used by said first set of instructions for processing the first incoming packet." Such claim elements, however, are allowable by virtue of the remarks above regarding Claim 4.

Further, with respect to Claim 30, applicant notes numerous deficiencies (including those set forth hereinabove regarding related claims). For example, with respect to applicant's claimed "wherein the apparatus includes a control logic unit coupled to an input and the policy condition table for feeding an arithmetic logic unit, which is in turn coupled to the policy action table and the state block for generating an output," the Examiner relies on the following excerpt from Abraham:

> "FIG. 4 is a block diagram of the component parts of the network management program 80 as distributed among the various computers and servers connected to the LAN 44. The GUI 70 of each administrative computer 54 and the network server 50 communicate the information and policies input by the operators of those computers to the rules and logging database 72 located on the network server 50 via the LAN 44. These policies are stored and processed by the rules and logging database 72, which then passes the user policies along to the filter executive 76 along with mapping information for each user. The filter executive 76 optimizes the policies into a set of rules for each user and passes the rules and user mapping information to the filter engine 78. The filter engine 78 filters all outbound IP packets transmitted from the LAN 44 to the Internet 40 and verifies all inbound IP packets from the Internet 40 according to the rules provided to the filter engine 78 by the filter executive 76. As this

occurs, the naming services manager 74 provides the filter executive 76 with updated mapping information which the filter executive then passes on to the filter engine 78 so that the filter engine begins and ceases filtering of IP packets dynamically as users log into and out of the LAN 44." (col. 9, lines 43 - 65)

Such excerpt, however, is replete with deficiencies. Just by way of example, the Examiner states that "Abraham discloses policies stored and processed by a rules and logging database" when addressing applicant's claimed "policy condition table for feeding an arithmetic logic unit." There is simply no arithmetic logic unit in Abraham, let alone a policy condition table feeding the same.

The foregoing excerpt simply fails to even suggest applicant's claimed "control logic unit coupled to an input and the policy condition table for **feeding an arithmetic logic unit, which is in turn coupled to the policy action table and the state block for generating an output**" (emphasis added).

Thus, all of the independent claims are deemed allowable. By virtue of their dependence on such independent claims, all of the remaining claims are further deemed allowable.

Reconsideration is respectfully requested.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 505-5100. For payment of the fees due in connection with the filing of this paper, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1351 (Order No. NAI1P069_99.074.01).

Respectfully submitted,
Silicon Valley IP Group, P.C.

Kevin J. Zilka
Registration No. 41,429

P.O. Box 721120
San Jose, CA 95172-1120
Telephone: (408) 505-5100